Ministerul Educaţiei Republicii Moldova

Universitatea Tehnică a Moldovei

Facultatea Calculatoare, Informatică şi Microelectronică.

Departamentul Informatică și Ingineria Sistemelor

Raport

la

Lucrare de laborator Nr.4

la Tehnici Avansate de Programare

**A efectuat: st. gr. IA-181 Paniș Iulian**

**A verificat: asist. univ Bumbu Tudor**

**Chișinau 2020Polimorfism**

**Scopul lucrării:**

1. Însuşirea modalităţilor de realizare a polimorfizmului în Java

**Etapele de realizare:**

1. Crearea clase noi;
2. Crearea metodelor necesare;
3. Realizarea formelor de polimorfizm;
4. Prezentarea lucrării

**Lucru individual:**

Polinom <- PoliniomArray

**Listingul programului:**

public class Main {

public static void main(String[] args) {

int arr1[] = {3, 5, 7, 0, 1};

int arr2[] = {2, 2, 9, 1};

int m = arr1.length;

int n = arr2.length;

//sumarea tablourilor

int arr3[] = Polynom.add(arr1, arr2, m, n);

//afisarea polinomului

System.out.println("Polinomul Nr1: ");

Polynom.show(arr3, arr3.length);

//utilizarea clasei secundare

double arr11[] = {0.22, 3.1, 5.67};

double arr22[] = {1.43, 0.74, 0.1, 3.9};

m = arr11.length; n = arr22.length;

//sumarea tablourilor

double arr33[] = PolynomArray.add(arr11, arr22, m, n );

//afisarea polinomului

System.out.println("Polinomul Nr2: ");

PolynomArray.show(arr33, arr33.length);

}

}

class Polynom{

//functia restituie valoarea maxima din 2 numere

public static int findMax(int m, int n) {

return (m > n) ? m : n;

}

//functia pentru adunare a 2 tablouri de tip int

static int[] add(int A[], int B[], int m, int n) {

//alocarea dimensiunii tabloului

int max = findMax(m, n);

int arr[] = new int[max];

//atribuirea primului tablou in cel final

for (int i = 0; i < m; i++) {

arr[i] = A[i];

}

//sumarea a 2 tablouri

for (int i = 0; i < n; i++) {

arr[i] += B[i];

}

return arr;

}

//functia pentru afisarea tabloului

static void show(int arr[], int n) {

for (int i = 0; i < n; i++) {

System.out.print(arr[i]);

//evitarea afisarii ultimului element sub forma X^1

if(i == n-2){

System.out.print("x + ");

continue;

}

//restul cazurilor

if (i != n - 1) {

System.out.print("x^" + (n - i-1)+" + ");

}

}

}

}

class PolynomArray extends Polynom{

//functia pentru adunare a 2 tablouri de tip double

static double[] add(double A[], double B[], int m, int n) {

//alocarea dimensiunii tabloului

int max = findMax(m, n);

double arr[] = new double[max];

//atribuirea primului tablou in cel final

for (int i = 0; i < m; i++) {

arr[i] = A[i];

}

//sumarea a 2 tablouri

for (int i = 0; i < n; i++) {

arr[i] += B[i];

}

return arr;

}

//functia pentru afisarea tabloului in sub alta forma

static void show(double arr[], int n) {

for (int i = 0; i < n; i++) {

System.out.print(arr[i]);

if (i != 0) {

System.out.print("x^" + i);

}

if (i != n - 1) {

System.out.print(" + ");

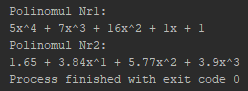
}

}

}

}

**Rezultatul:**

****

**Concluzii:**

În urma lucrării de laborator Nr.4 am realizat în practică specificațiile ***polimorfismului.***

Am creat 2 clase noi (Superclasa->Subclasa), am creat metodele necesare în superclasă, ulterior modificându-le în subclasă după tipul de date preluat și returnat.După aceasta fiecare metodă în parte a fost utilizată în cadrul clasei de bază **Main**, prin specificarea clasei la care aparține în dependență de tipul de date ale tablourilor.